v1 Explicação

Etapa 1: Análise exploratória de dados (EDA)

1.0 Importar dados e bibliotecas

- Importar `numpy`, `pandas`, `matplotlib`, `seaborn` para processamento e visualização de dados.
- Preencher os valores em falta com `SimpleImputer` em `sklearn.impute`.
- Carregue o conjunto de dados `custom_covid19.csv` e obtenha uma ideia preliminar do formato dos dados, do número de colunas, do tipo de dados e da ausência de dados com `df.head()` e `df.info()`.

1.1 Construir variáveis-alvo e classificar caraterísticas

- Construa uma nova coluna `DIED` (0 = não morto, 1 = morto) determinando se o campo DATE_DIED é `9999-99-99`.
- Classificar as caraterísticas pelo número de valores únicos na coluna com o tipo de dados:
  - `cat_cols`: variável de categoria (<= 20 valores únicos, excluindo a coluna de destino e a data)
  - `num_cols`: variáveis numéricas (int/float e não em `cat_cols`)

1.2 Tratamento dos valores em falta

- Todos os valores especiais em falta (97, 98, 99) são uniformemente substituídos por `np.nan`.
- As variáveis categóricas são preenchidas com os valores mais frequentes (moda); as variáveis numéricas são preenchidas com a mediana, o que é mais eficiente do que o KNN.
- Isto pode evitar erros de comunicação devido a valores em falta durante o treino do modelo.

1.3 Deteção de valores anómalos

- Apenas são detectados os valores atípicos na `AGE`, utilizando o método IQR (1,5 vezes os quartis superior e inferior).

- Os limites superior e inferior e o número de valores atípicos são impressos e visualizados através de gráficos de caixa, mas não são eliminados, uma vez que o modelo de árvore é mais robusto em relação aos valores atípicos.

1.4 Visualização da distribuição de cada variável

- Trace as variáveis `IDADE`, `SEXO`, `TIPO DE PACIENTE` e `UCI` separadamente:
  - `Histplot (histograma`) é utilizado para valores numéricos.
  - `countplot` para o tipo de categoria.
- O objetivo é compreender as caraterísticas das distribuições enviesadas e de classe das variáveis.

1.5 Mapa de calor da correlação

- Calcular os coeficientes de correlação de Pearson para todas as caraterísticas numéricas.
- Demonstrar com o mapa de calor: AGE está fracamente correlacionada (<0,25) com a maioria das variáveis, TEST_RESULT quase não está correlacionada, sugerindo que O2 e O3 são difíceis.

1.6 Seleção manual de caraterísticas + teste do qui-quadrado

- A lista de variáveis importantes para cada tarefa `important_features_O1/O2/O3` foi selecionada com base no conhecimento médico comum.
- Foi efectuado o teste do qui-quadrado com DIED para todas as variáveis categóricas e foram listadas as variáveis significativas (p<0,05).

1.7 Comparação das médias dos grupos

- As médias das caraterísticas foram calculadas e visualizadas para os grupos DIED=0 e DIED=1, respetivamente, e verificou-se que UCI, INTUBADO, PNEUMONIA, etc., eram, em média, mais elevadas no grupo de morte.

1.8 Distribuição das categorias

- Estatísticas e visualização da distribuição de categorias de DIED: apenas cerca de 7% das pessoas mortas, o que indica que a amostra está

seriamente desequilibrada, pelo que o modelo de classificação precisa de adicionar class_weight='balanced'.

1.9 Guardar os dados limpos

- As colunas que são numéricas mas deviam ser inteiras (por exemplo, AGE) são uniformemente convertidas para o tipo Int64.
- Por fim, guarde os dados limpos como `custom_covid19_cleaned.csv`.

Passo 2: Tarefa O1 - Previsão de morte (classificação)

2.1 Preparação dos dados

- Utilize as 10 variáveis selecionadas pela Tarefa O1 como caraterísticas que visam `DIED`.
- Dividir os conjuntos de treino e validação, estratificar=y para garantir que as distribuições das duas classes são consistentes.

2.2 Construção do pipeline de modelação

São construídos três pipelines de modelos:

- Regressão logística: com escalonamento (StandardScaler) e pesos de classe equilibrados
- Random Forest: sem escalonamento, modelação direta
- MLP: mais escalonamento, hidden_layer = (20,)

2.2.1 Afinação de hiperparâmetros

- Para a Regressão Logística, ajustar a força regular `C`
- Ajustar o tamanho da camada oculta e `o alfa` para MLP
- Utilizar a validação cruzada 5 vezes + pontuação f1 para encontrar o modelo ótimo
- RF Utilizar diretamente os parâmetros predefinidos (n=200)

2.3 Avaliação do conjunto de validação

- Avaliar o desempenho de cada modelo no conjunto de validação, as métricas incluem:

- F1, Recall: capacidade de direcionar para classes positivas (fatalidades)
- MCC: tratamento mais robusto dos desequilíbrios
- AUC: medida da capacidade de discriminar entre probabilidades previstas

2.4 Visualização dos resultados

- Visualização dos resultados da previsão utilizando a matriz de confusão.

2.5 Teste de conjuntos de teste independentes

- Carregar `proj-test-data.csv` e `proj-test-class.csv`
- Teste diretamente o `melhor modelo` e calcule as métricas de classificação e a MCC.

2.6 Salvar o melhor modelo

- Guarde o modelo com melhor desempenho (com o MCC mais elevado) como `best_death_classifier.pkl`.

🎯 Passo 3: Tarefa O2 - Previsão da idade (regressão)

3.1 Preparação dos dados

- Caracterizar 9 variáveis de classe de doença (0/1)
- Dividir os conjuntos de treino e validação, pré-processar com ColumnTransformer: OneHotEncoder + StandardScaler

3.2 Construção do modelo

- Quatro modelos: Random Forest, HistGBR, Ridge, Linear.
- O HistGBR é o modelo principal, de formação rápida e desempenho estável.

3.3 Afinação de RF

- Utilizar GridSearchCV para ajustar n_estimators e max_depth, validação cruzada 5 vezes, o melhor modelo é best_rf.

3.4 Ajuste do modelo

- Treinar todos os modelos e preparar a avaliação uniforme.

3.5 Avaliação do conjunto de validação

- As métricas incluem: MAE, RMSE, R^2 e correlação de Pearson.
- O HistGBR tem o melhor desempenho: MAE ≈ 11,75, R^2 ≈ 0,2177

3.6 Guardar o melhor modelo

- Guardar HistGBR como `histgbr_age_predictor.pkl`

3.7 Visualização do preditor vs. real

- Gráfico de dispersão + Linha de referência ideal

3.8 Avaliação do conjunto de teste independente

- Os resultados no conjunto de teste são quase idênticos aos do conjunto de validação, indicando que o modelo generaliza bem.

3.9 Validação cruzada adicional (a ser apresentada apenas na defesa)

- Resultados da validação cruzada 5 vezes MAE ±0,08 e RMSE ±0,10, indicando que o modelo é estável.

Passo 4: Tarefa O3 - Previsão da idade do falecido (regressão)

4.1 Preparação dos dados.

- Triagem da amostra de óbitos com DIED=1
- Caracterizada por variáveis importantes semelhantes a doenças, 11 no total
- O objetivo é a IDADE
- Dividir os conjuntos de treino e validação, preencher as lacunas

4.2 Modelação

- Utilizar RandomForest e GradientBoosting com parâmetros predefinidos.

4.3 Avaliação da validação

- Os resultados globais são fracos: R^2 ≈ 0,10, indicando que a variável tem pouco ou nenhum poder preditivo para a idade

4.4 Validação cruzada

- Confirmação adicional de que o modelo é instável e tem um erro elevado

4.5 Guardar o modelo

- Embora o modelo não seja recomendado, o GradientBoosting ainda é guardado como uma demonstração do processo.

4.6 Avaliação do conjunto de teste

- O R^2 negativo no conjunto de teste independente indica que o modelo é pouco eficaz

4.7 Resumo

- O modelo tem um desempenho muito fraco porque:
  - Tamanho reduzido da amostra
  - Baixo conteúdo informativo das caraterísticas
  - A idade em si está fracamente correlacionada com outras variáveis

---

Etapa 5: Resumo das caraterísticas e reflexão sobre o projeto

5.1 Comparação das caraterísticas importantes

- Tabela que lista as variáveis importantes para cada tarefa

5.2 Resumo do impacto das caraterísticas

- O1: O indicador de doença grave é a principal causa de morte
- O2: A doença está fracamente correlacionada com a idade
- O3: As caraterísticas da doença não conseguem prever a idade

5.3 Resumo do projeto

- Um passeio completo pelo fluxo de trabalho da aprendizagem automática
- O modelo O1 funciona melhor; O2 tem algum erro; O3 não pode ser modelado.
- Foram utilizadas ferramentas para ajudar, mas a compreensão foi o principal objetivo.